



## Dictionary Based Segmentation in Volumes

**Emerson, Monica Jane; Jespersen, Kristine Munk; Jørgensen, Peter Stanley; Larsen, Rasmus; Dahl, Anders Bjørholm**

*Published in:*  
Image Analysis

*Link to article, DOI:*  
[10.1007/978-3-319-19665-7\\_43](https://doi.org/10.1007/978-3-319-19665-7_43)

*Publication date:*  
2015

*Document Version*  
Peer reviewed version

[Link back to DTU Orbit](#)

*Citation (APA):*  
Emerson, M. J., Jespersen, K. M., Jørgensen, P. S., Larsen, R., & Dahl, A. B. (2015). Dictionary Based Segmentation in Volumes. In *Image Analysis: 19th Scandinavian Conference, SCIA 2015 Copenhagen, Denmark, June 15–17, 2015 Proceedings* (pp. 504-515). Springer Science+Business Media. Lecture Notes in Computer Science [https://doi.org/10.1007/978-3-319-19665-7\\_43](https://doi.org/10.1007/978-3-319-19665-7_43)

---

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# Dictionary Based Segmentation in Volumes

Monica Jane Emerson<sup>1</sup> (✉), Kristine Munk Jespersen<sup>2</sup>, Peter Stanley Jørgensen<sup>3</sup>, Rasmus Larsen<sup>1</sup>, and Anders Bjorholm Dahl<sup>1</sup>

<sup>1</sup> DTU Compute, Lyngby, Denmark

<sup>2</sup> DTU Wind Energy, Roskilde, Denmark

<sup>3</sup> DTU Energy, Roskilde, Denmark

monj@dtu.dk, kmun@dtu.dk, psjq@dtu.dk, rlar@dtu.dk, abda@dtu.dk

**Abstract.** We present a method for supervised volumetric segmentation based on a dictionary of small cubes composed of pairs of intensity and label cubes. Intensity cubes are small image volumes where each voxel contains an image intensity. Label cubes are volumes with voxel-wise probabilities for a given label. An unknown volume is segmented by cube-wise finding the most similar dictionary intensity cube. From these matches we use the probabilities from the label cubes to build a probability volume where overlapping cubes are averaged to obtain a robust label probability encoding. The dictionary is computed from labeled volumetric image data based on weighted clustering. We experimentally demonstrate our method using two data sets from material science – phantom data of solid oxide fuel cell simulations for detecting three phases, and a tomogram of a glass fiber composite used in wind turbine blades for detecting individual glass fibers.

**Keywords:** Volume segmentation, Materials images, X-ray tomography, Learning dictionaries, Glass fiber segmentation

## 1 Introduction

High resolution volumetric scanning has become a widely used technique in areas like material science and medicine, and automated quantification methods are necessary in order to obtain size and shape measures from these data. We present a method for supervised segmentation of volumetric data. The method is trained from manual annotations, and these annotations make the method very flexible, which we demonstrate in our experiments.

Our method infers label information locally by matching the pattern in a neighborhood around a voxel to a dictionary, and hereby accounts for the volume texture. Texture segmentation has been widely addressed in 2D [4, 8, 19], whereas volumetric texture segmentation has received less attention [1]. This fact could be due to the extra computational effort introduced in 3D. However, 3D texture segmentation is highly appropriate for quantifying size and shape in 3D data. Applications in this paper are from energy material science.

Denmark is pursuing to obtain 100% of the energy from renewable resources by 2050 [6]. For this reason, it is important to develop renewable energy technologies, which involves investigating material properties to ensure efficiency and

lifetime. Imaging the 3D micro-structure can be essential in characterizing and understanding such properties, e.g. the geometric configuration of a solid oxide fuel cell or the fiber geometry in a wind turbine blade material, which are the applications we investigate here.

Our applications contain three spatial dimensions, but data could also be two spatial dimensions and time as the third. If it were to be the latter, the evolution of the material's micro-structure under certain conditions (e.g. temperature or tension) could be investigated. Moreover, the flow of different elements (e.g. gas or liquid) through porous materials can be investigated [17]. Other applications are segmentation of anatomical structures in medical imaging [1, 16], seismic facies analysis [14] or crystallography [15]. Many of the volumetric texture segmentation algorithms are extensions of common 2D techniques [2] or 2D segmentation propagation approaches for segmenting 3D materials [18].

Wind turbines blades commonly use glass fiber composites for the load carrying parts of the blades, for which the fatigue damage mechanisms are not well understood. Wind turbine blades have long expected lifespans where they experience a high number of load cycles, which gives rise to fatigue damage evolution. In addition, the blade lengths are being increased because the power output of a wind turbine is proportional to the blade length squared. As fatigue is one of the main limiting factors of designing longer blades, improving the understanding of fatigue damage evolution in glass fiber composites is important [13, 7]. Here microstructure analysis using imaging is an important tool, both for material characterization and modeling.

Solid oxide fuel cells (SOFC) operate by oxidizing a fuel to produce electricity and heat. The electrodes of an SOFC are typically two-phase porous systems. The two solid phases are responsible for electron and ion conduction and the pores allow transport of gaseous reactants and products to and from the electrochemically active sites at the triple phase boundaries (TPB). The chemical reactions can only take place at the TPBs where there is access for electrons, ions and gases through the corresponding three phases. The performance of an SOFC is thus strongly dependent on the density of TPBs in the electrodes and on how easily, electrons ions and gases can be transported to and from the TPBs [10]. The microstructure of an SOFC can only be indirectly controlled through a complicated interaction between powder particle sizes, casting methods and sintering temperatures. 3D characterization of the micro-structure is thus becoming an increasingly important tool to correlate the characteristics of the micro-structure to the cell performance and the production recipes.

In this paper we extend the 2D segmentation algorithm in [5] to 3D. The method is based on a dictionary of image patches and corresponding labels patches. Here we replace the image patches with volume cubes and investigate the effects on the segmentation. We see an improved performance in some of our experiments by extending to 3D, but at the expense of longer computation times. However, we obtain close to perfect segmentation of individual glass fibers in wind turbine blades, and also high performance in segmenting solid oxide fuel cell data.

The paper is organized as follows. In Section 2, we provide a description of the algorithm. Section 3 explains the data sets and materials which are utilized for the comparative study. In Section 4, the focus is on the results, where three methods (method from literature, 2D dictionary and 3D dictionary) are compared for each data set. Finally, Section 5 draws some conclusions.

## 2 Method

In the process of extending [5] to 3D, we have however changed some parts and therefore we include a description of the entire method despite the overlap with [5]. The method is based on a dictionary of small intensity cubes coupled with label cubes, and the dictionary is learned using weighted clustering. First we describe the dictionary, then we explain the problem we are optimizing, and finally we describe an algorithm to compute the dictionary and the method for inferring label information to an unlabeled volume.

The dictionary is based on annotated training data, where the annotation assigns each voxel in the training volume to one class label. Given a volume  $V_I : \Omega \rightarrow \mathbb{R}$  where  $\Omega \subset \mathbb{R}^3$  and an annotation of that volume  $V_L : \Omega \rightarrow \mathbb{N}$  with labels  $l = 1, \dots, k$ , we want to build a dictionary  $\mathbf{D} = (\mathbf{D}_I, \mathbf{D}_L)$  consisting of the intensity dictionary  $\mathbf{D}_I \in \mathbb{R}^{m \times n}$  and the associated label dictionary  $\mathbf{D}_L \in \mathbb{R}^{km \times n}$ . Each column in  $\mathbf{D}_I$  contains a vector representation of small intensity cubes of side length  $M$  where  $m = M^3$ , and  $n$  is the number of dictionary elements. The columns of  $\mathbf{D}_L$  contain vectorized label cubes represented as probabilities of labels. Therefore, each label vector contains  $km$  elements, where the first  $m$  elements are the probabilities of label 1, the next  $m$  elements are for label 2, etc. We get

$$\sum_{l=0}^{k-1} \mathbf{d}_{Li}(lm + \tau) = 1 \text{ for all } \tau \in \{1, \dots, m\},$$

where  $\mathbf{d}_{Li}$  is column  $i$  in  $\mathbf{D}_L$ .

An ideal dictionary would be

$$\hat{\mathbf{D}} = \arg \min_{\mathbf{D}} \sum_{\eta=1}^o \left( \hat{\lambda} \|\mathbf{d}_{Li(\eta)} - \mathbf{v}_{L\eta}\|_2^2 + \|\mathbf{d}_{Ii(\eta)} - \mathbf{v}_{I\eta}\|_2^2 \right), \quad (1)$$

where  $\mathbf{v}_{I\eta}$  and  $\mathbf{v}_{L\eta}$  are the  $\eta$ 'th intensity cube and label cube from  $V_I$ ,  $V_L$  respectively with  $\eta = 1, \dots, o$  where  $o$  is the number of intensity cubes in  $V_I$  and  $\hat{\lambda}$  is a scaling factor.  $i(\eta)$  is the index of the nearest intensity dictionary element

$$i(\eta) = \arg \min_i \|\mathbf{d}_{Ii} - \mathbf{v}_{I\eta}\|_2^2. \quad (2)$$

The first norm in (1)  $\|\mathbf{d}_{Li(\eta)} - \mathbf{v}_{L\eta}\|_2^2$  measures the Euclidean distance between the label dictionary element, and label cube and the second norm in (1)  $\|\mathbf{d}_{Ii(\eta)} - \mathbf{v}_{I\eta}\|_2^2$  is the Euclidean distance between the intensity dictionary element and intensity cube. Our aim is to minimize both simultaneously because we

hereby obtain a discriminative dictionary and good clustering properties. We will later show that this is advantageous when using the dictionary for segmenting an unlabeled image.

Finding a solution to (1) is a hard problem. Without the first norm concerning the labels it is a  $k$ -means clustering problem, which is NP-hard, and we have not found a solution with the addition of the label information. So, based on (1) we suggest a heuristic clustering algorithm which has given good performance in our experiments. The basic idea is to estimate a weight from the label information and use that in a weighted  $k$ -means clustering approach.

The algorithm for building the dictionary is based on iteratively updating a set of dictionary elements. Initially a random set of  $\xi$  associated intensity and label cubes are selected as  $\hat{\mathbf{v}}_j = (\hat{\mathbf{v}}_{Ij}, \hat{\mathbf{v}}_{Lj})$  from the annotated training volume and vectorized, where  $j = 1, \dots, \xi$ . A subset of  $n$  patches are randomly selected as the initial dictionary  $\mathbf{D}^0$ , where  $n < \xi$ . New dictionary elements are now estimated iteratively as

$$\mathbf{d}_{Ii}^{t+1} = \frac{1}{v_i} \sum_{\kappa \in S_i} (1 - \lambda \|\mathbf{d}_{Li}^t - \hat{\mathbf{v}}_{L\kappa}\|_2) \hat{\mathbf{v}}_{I\kappa}, \quad (3)$$

where  $\mathbf{d}_{Ii}^{t+1}$  is the intensity dictionary element at iteration number  $t + 1$ .  $S_i$  is the set of indices with intensity cubes closest to dictionary element  $i$

$$\kappa \in S_i \text{ s.t. } \kappa = \arg \min_j \|\mathbf{d}_{Ii} - \mathbf{v}_{Ij}\|_2^2.$$

The normalization factor is estimated as

$$v_i = \sum_{\kappa \in S_i} (1 - \lambda \|\mathbf{d}_{Li} - \hat{\mathbf{v}}_{L\kappa}\|_2). \quad (4)$$

This approach gives high weight to training samples with labels similar to the dictionary element and low weight to dissimilar samples. The label dictionary elements are estimated as average labels for the cluster as

$$\mathbf{d}_{Li}^{t+1} = \frac{1}{|S_i|_0} \sum_{\kappa \in S_i} \hat{\mathbf{v}}_{L\kappa}, \quad (5)$$

where  $|S_i|_0$  is the cardinality of  $S_i$ . In our experiments we have seen a satisfactory result with little change in the dictionary after approximately 10 iterations.

The dictionary is used for segmenting a volume by building a label probability volume. This is done by matching the intensity dictionary elements to the volume we want to segment, and adding the associated label dictionary elements to an empty label probability volume.

Given a volume  $U_I : \Omega \rightarrow \mathbb{R}$  that we want to segment with  $U_I \in \mathbb{R}^{x \times y \times z}$  we compute a label probability volume  $U_L : \Omega \rightarrow \mathbb{R}$  with  $U_L \in \mathbb{R}^{x \times y \times z \times k}$ . Initially we set  $U_L$  to having all elements zeros. We can extract vectorized intensity cubes  $\mathbf{u}_{Ih}$  of the same spatial size as the dictionary elements, where  $h = 1, \dots, \rho$  is the number of possible cubes with side length  $M$ , so e.g. for an odd  $M$  we get

$\rho = (x - M + 1)(y - M + 1)(z - M + 1)$ , which is a little less than the number of voxels due to the volume boundaries. Each  $\mathbf{u}_{Ih}$  is matched to the nearest intensity dictionary element using Euclidian distance. For a given intensity vector  $\mathbf{u}_{Ih}$  we get the nearest dictionary element

$$i(h) = \arg \min_i \|\mathbf{d}_{Ii} - \mathbf{u}_{Ih}\|_2^2. \quad (6)$$

From this we take the corresponding label dictionary element  $\mathbf{d}_{Li(i(h))}$  and add it to the label volume  $U_L$  at the coordinates of  $h$ 'th cube extracted from  $U_I$  for each of the  $k$  labels. When the probabilities are added we weigh them using a Gaussian weight function with standard deviation  $\sigma$  centered at the cube. After adding the probabilities in  $U_L$  up, we simply normalize by dividing the sum over label probabilities for each voxel to make the probabilities sum to one over all  $k$  labels.

Some smoothing at boundaries occurs which especially affects small features. In order to account for that, we estimate label-wise weights on an annotated validation set where we minimize the difference between the obtained probability volume and the annotation. We are given an annotated volume  $Q_L$  and the computed label probability volume  $P_L$  using a trained dictionary. Then we rearrange these volumes to  $\mathbf{Q}_L$  and  $\mathbf{P}_L$  such that each row contains the voxel-wise probabilities and each column represents the labels with each row summing to 1, i.e.  $\mathbf{Q}_L(r, l) \in \{0, 1\}$ ,  $\mathbf{P}_L(r, l) \in [0, 1]$  and  $\sum_{c=1}^k \mathbf{Q}_L(r, l) = \sum_{c=1}^k \mathbf{P}_L(r, l) = 1$ , where  $\mathbf{Q}_L(r, l)$  and  $\mathbf{P}_L(r, l)$  are elements from  $\mathbf{Q}_L$  and  $\mathbf{P}_L$  at row  $r$  and column  $l$  respectively. We want to find the weight matrix  $\mathbf{W} \in \mathbb{R}^{k \times k}$  that minimizes

$$\mathbf{W} = \arg \min_{\mathbf{W}} \|\mathbf{Q}_L - \mathbf{P}_L \mathbf{W}\|_2^2,$$

where the solution is found as

$$\mathbf{W} = (\mathbf{P}_L^T \mathbf{P}_L)^{-1} \mathbf{P}_L^T \mathbf{Q}_L.$$

The voxel-wise probability of the final segmentation is obtained as

$$\tilde{\mathbf{u}}_L(\mathbf{x}) = \mathbf{u}_L(\mathbf{x}) \mathbf{W}, \quad (7)$$

where  $\mathbf{u}_L(\mathbf{x}) \in \mathbb{R}^k$  is a vector of label probabilities of the voxel from the spatial position  $\mathbf{x} = (x, y, z)^T$  in the intensity volume  $U_I$ .

### 3 Materials

Two data sets are employed for the comparative study including:

1. Real data from glass fiber used for wind turbine blades.
2. Phantom data of solid oxide fuel cells.

### 3.1 Glass Fiber

The data set was obtained through 3D X-ray computed tomography imaging. The scanned sample is a cut-out of a fatigue test specimen and the dimensions of approximately  $5 \times 5 \times 10$  mm. The material considered is a uni-directional (UD) glass fiber/polyester composite used in the load carrying beam of a wind turbine blade. Uni-directional in this case means that the fibers are aligned in one main direction, making the composite strong in one direction, and weak in other directions. In order to hold the UD fiber bundles in place during manufacturing, they are stitched to a thin layer of transverse backing fiber bundles. As the backing only contributes lightly to the mechanical properties of the material, the main focus in this study is on segmenting the UD fibers.

In Figure 1, we see one of the training slices and its corresponding annotated labeling where three different classes are defined: centers, fibers and matrix.

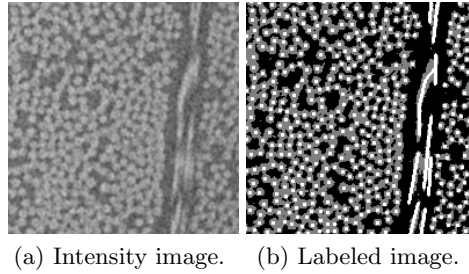


Fig. 1: One slice of the fiber training data. In *white*: centers, *gray*: fibers and *black*: matrix.

### 3.2 Solid Oxide Fuel Cell Phantom

We simulated a 3D SOFC phantom using a random process employing a simple curvature minimization scheme. Here the target phase fractions of 0.24, 0.38 and 0.38 were used for the pore, ion conducting and electron conducting phases respectively. This phantom creation scheme greatly simplifies many aspects of the evolution of a real micro-structure during sintering and as such the created micro-structure is not suited for comparison of higher order properties to a real SOFC. However, the scheme provides 3D micro-structure data that qualitatively matches the structure observed in real SOFC data sets [9]. The scheme is thus well suited as ground truth for segmentation. After the phantom creation of the artificial 3D micro-structure data the X-ray projections and reconstruction were simulated using slice wise radon and inverse radon transform using AIR tools<sup>4</sup>. 3% Gaussian noise was added to the radon transformed data before reconstructing it.

In Figure 2, we can see one of the training slices and its corresponding annotated labeling, where three classes have been defined, corresponding to each of the three phases.

<sup>4</sup> <http://www2.compute.dtu.dk/~pcha/AIRtools/>

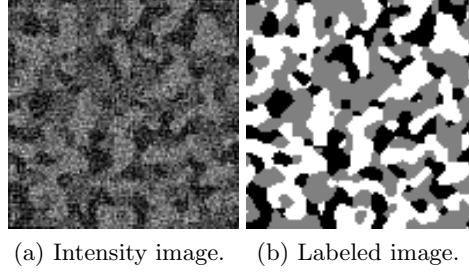


Fig. 2: One slice of the fuel cell training data. In *white*: phase one, *gray*: phase two and *black*: phase three.

## 4 Results

In this section we provide the results from our algorithm and compare it to well established image analysis methods. Glass fiber detection is compared to 2D scale space blob detection [12] and SOFC segmentation is compared to Markov random field (MRF) segmentation solved using graph cuts with alpha expansion [3, 11]. For both methods we chose parameters that minimized the segmentation error. It should be noted that the parameters for the MRF are not learned, i. e. they are known.

For the dictionary algorithms, several parameters need to be set. It is known that the segmentation is most sensitive to the change in atom size  $M$ , i.e. the side length of the cubes, and number of dictionary elements  $n$ , so segmentation errors will be calculated for different parameter settings so as to select the optimal.

### 4.1 Glass Fiber

For the glass fiber, the dictionaries have been trained with 9 slices of size  $200 \times 200$  pixels, the validation set also contains 9 slices of the same size. The performance is calculated over only one test slice of size  $500 \times 500$  pixels.

The performance measure is computed object-wise over the center class, and represents the true positive rate and false positive rate. We segment the center part of the fibers, and the individual fibers are found as the center of mass using connected component analysis on the center segments. Comparison to the manual marked ground truth is done by counting the number of true matches, which are found as points that have each other as nearest neighbors as well as a distance of less than 4 voxels. The 4 voxels are chosen because the average radius of the glass fibers is around 4 voxels. Results are shown in Table 1. The computational time, defined as the time to train, validate and classify the one test slice, has also been calculated.

The computational time ( $C_{time}$ ) grows as the number of elements in the dictionary increases ( $n$ ) or the size of the volume cube ( $M$ ) becomes bigger.



Table 1: Performance measure dictionaries with different parameters.

$M$	$n$	TPR <sup>2D</sup>	FPR <sup>2D</sup>	TPR <sup>3D</sup>	FPR <sup>3D</sup>
3	50	0.8563	0.1674	0.9099	0.1202
3	250	0.9082	0.1678	0.9511	0.1520
3	1000	0.9350	0.1721	0.9561	0.1165
5	50	0.9468	0.0399	0.9926	0.0177
5	250	0.9762	0.0248	0.9977	0.0104
5	1000	0.9799	0.0208	0.9963	0.0121
7	50	0.9866	0.0104	0.9896	0.0104
7	250	0.9873	0.0117	0.9973	<b>0.0080</b>
7	1000	0.9910	0.0144	<b>0.9980</b>	0.0087
9	50	0.9484	0.0171	0.9461	0.0107
9	250	0.9930	0.0100	0.9940	0.0104
9	1000	<b>0.9953</b>	<b>0.0097</b>	0.9977	0.0100

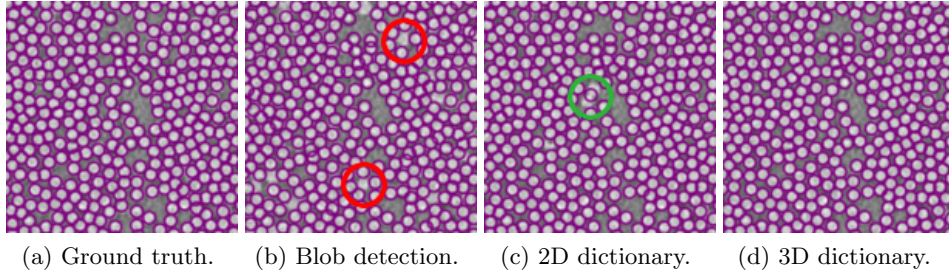


Fig. 3: Ground truth and circles around detected fiber centers.

The average  $C_{time}$  is 100.3 seconds for 3D, whereas it is 3.2 seconds for the 2D dictionary.

In Figure 3, we see a zoomed image of fibers with circles around each of the detected centers for each of the three methods (blob detection, 2D dictionary and 3D dictionary) and the ground truth image. Circles are plotted with radius 4 for illustration purposes. As can be seen qualitatively from the images, the performance of the dictionary methods is substantially better than the one obtained through blob detection. The centers in the blob detection are found less accurately compared to those in the ground truth and some centers are not found (e.g. red circles in Figure 3). The displacement of the centers in the 2D and 3D dictionary with respect to the ground truth is very small. Moreover, the 3D dictionary performs slightly better than the 2D dictionary, as it finds all the centers, whereas in the 2D method results there are two centers which have been detected as one only center (blue circle Figure 3). This is because there are some pixels connecting these two centers.

Quantitatively the three methods can be compared using the performance measures TPR and FPR.

1. **Blob detection:** TPR = 0.8339 and FPR = 0.0737.
2. **2D dictionary** ( $M = 9$ ,  $n = 1000$ ): TPR = 0.9953 and FPR = 0.0097.
3. **3D dictionary** ( $M = 7$ ,  $n = 1000$ ): TPR = 0.9980 and FPR = 0.0087.

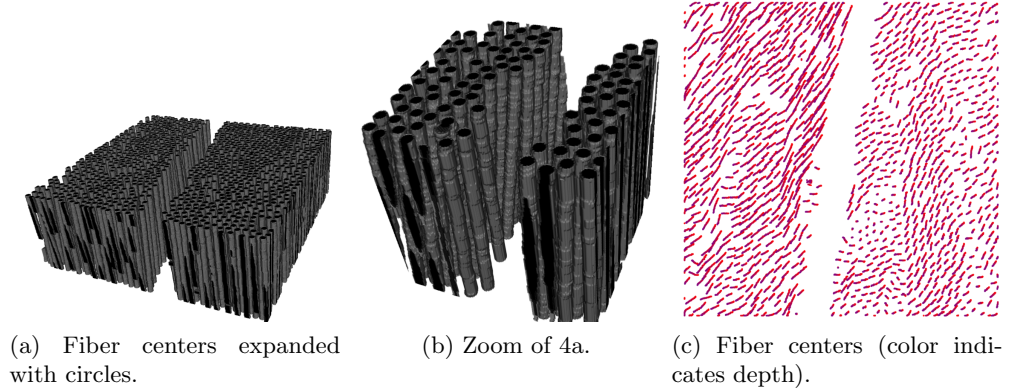


Fig. 4: Segmentation over a stack of 100 slices.

In Figure 4, a stack of 100 slices of  $300 \times 300$  pixels was segmented using the 3D dictionary ( $M = 7$ ,  $n = 1000$ ). In these images we present the results in 3D, where the fiber orientation can be visualized (straight fibers appear as points in 4c). Note how accurate the fiber orientation can be visualized using this approach.

#### 4.2 Solid Oxide Fuel Cell Phantom

For the fuel cell data, the measure used to evaluate the performance is defined as the percentage of pixels which are classified correctly over all three classes. Results are shown in Table 2. The dictionaries are trained with a volume of  $50 \times 100 \times 100$  pixels, validated with another volume of  $50 \times 100 \times 100$  and the performance is measured over a volume of  $100 \times 100 \times 100$  voxels.

Table 2: Performance measure dictionaries with different parameters.

$M$	$n$	Performance <sup>2D</sup>	Performance <sup>3D</sup>
3	50	0.8864	0.9115
3	250	0.8917	<b>0.9197</b>
3	1000	0.8913	0.9192
5	50	0.9029	0.8886
5	250	0.9103	0.9034
5	1000	<b>0.9119</b>	0.9060
7	50	0.8819	0.8227
7	250	0.8995	0.8453
7	1000	0.9069	0.8512
9	50	0.8407	0.7455
9	250	0.8743	0.7786
9	1000	0.8886	0.7803

As for the glass fiber, the computational time is two orders greater when using the 3D dictionary. In Figure 5, we see one segmented slice from the test set for each of the three methods (MRF, 2D dictionary and 3D dictionary) compared to the ground truth image. In this case, the benefit from the 3D expansion of the

dictionary method is not significant due to the excessive smoothing introduced by the 3D dictionary. However, the dictionary methods do slightly outperform the MRF technique, as we can see from in the following quantitative results:

1. **MRF**: Performance = 0.9078.
2. **2D dictionary** ( $M = 5$ ,  $n = 1000$ ): Performance = 0.9119.
3. **3D dictionary** ( $M = 3$ ,  $n = 250$ ): Performance = 0.9197.

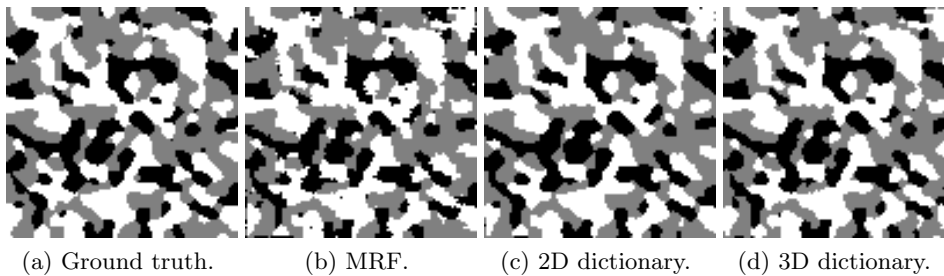


Fig. 5: Ground truth and segmentations of fuel cell phantoms.

## 5 Conclusions

In this paper, we have presented a highly flexible and accurate method for 3D segmentation of complex image structures. The method is an extension of a 2D segmentation method and we have investigated if adding a new spatial dimension improved the segmentation performance, as this third dimension provides extra contextual information. We have conducted an investigation to demonstrate the flexibility of the proposed method. Our investigation included experiments for segmenting individual glass fibers in X-ray computed tomography data and for segmenting three phases in phantom data of solid oxide fuel cell data. With the glass fiber data we obtained close to perfect segmentation of the fiber centers. For the fuel cell phantoms we almost see no improvement going from 2D to 3D, which might be due to the excessive smoothing introduced by the 3D algorithm. The fiber detection was compared to scale space blob detection and the solid oxide fuel cell data was compared to Markov random field segmentation using alpha expansion. In both cases both the 2D and 3D methods outperformed these standard methods.

In the current implementation the computational cost is high, but the method has not been optimized for speed. Especially the 3D version is computationally expensive with two orders of magnitude higher computation time and our plan to optimize the method for computational speed in our future work. With the ease of training and very high performance, we believe that the proposed methods can be a very useful tool in quantifying structures in complex volumetric data like tomograms of material samples.

## References

1. Aldasoro, C.C.R., Bhalerao, A.: Volumetric texture segmentation by discriminant feature selection and multiresolution classification. *IEEE TMI* 26(1), 1–14 (2007)

2. Blot, L., Zwiggelaar, R.: Synthesis and analysis of solid texture: application in medical imaging pp. 9–12 (2002)
3. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. *IEEE TPAMI* 23(11), 1222–1239 (2001)
4. Cremers, D., Rousson, M., Deriche, R.: A review of statistical approaches to level set segmentation: integrating color, texture, motion and shape. *International Journal of Computer Vision* 72(2), 195–215 (2007)
5. Dahl, A.L., Larsen, R.: Learning dictionaries of discriminative image patches. In: 22nd BMVC (2011)
6. Government, T.D.: The danish climate policy plan – towards a low carbon society. Tech. rep., Danish Energy Agency (2013)
7. Hansen, J.Z., Brøndsted, P., Jacobsen, T.K.: The effects of fibre architecture on fatigue life-time of composite materials. Ph.D. thesis, Technical University of Denmark, Risø National Laboratory for Sustainable Energy (2013)
8. Ilea, D.E., Whelan, P.F.: Image segmentation based on the integration of colour–texture descriptors a review. *Pattern Recognition* 44(10), 2479–2501 (2011)
9. Jørgensen, P.S., Yakal-Kremski, K., Wilson, J., Bowen, J.R., Barnett, S.: On the accuracy of triple phase boundary lengths calculated from tomographic image data. *Journal of Power Sources* 261, 198 – 205 (2014)
10. Jørgensen, P., Ebbenhøj, S., Hauch, A.: Triple phase boundary specific pathway analysis for quantitative characterization of solid oxide cell electrode microstructure. *Journal of Power Sources* 279, 686 – 693 (2015)
11. Kolmogorov, V., Zabih, R.: What energy functions can be minimized via graph cuts? *IEEE TPAMI* 26(2), 147–159 (2004)
12. Lindeberg, T.: Feature detection with automatic scale selection. *International Journal of Computer Vision* 30(2), 79–116 (1998)
13. Nijssen, R.P.L.: Fatigue life prediction and strength degradation of wind turbine rotor blade composites. Contractor Report SAND2006-7810P, Sandia National Laboratories, Albuquerque, NM (2006)
14. Randen, T., Monsen, E., Signer, C., Abrahamsen, A., Hansen, J.O., Sæter, T., Schlaf, J., Sønneland, L., et al.: Three-dimensional texture attributes for seismic data analysis. In: 70th Annual International Meeting, Society of Exploration Geophysics Expanded Abstracts. pp. 668–671 (2000)
15. Tai, C., Baba-Kishi, K.: Microtexture studies of pst and pzt ceramics and pzt thin film by electron backscatter diffraction patterns. *Textures and Microstructures* 35(2), 71–86 (2002)
16. Tu, Z., Zhou, X.S., Comaniciu, D., Bogoni, L.: A learning based approach for 3d segmentation and colon detagging. In: *Computer Vision–ECCV 2006*, pp. 436–448. Springer (2006)
17. Ushizima, D., Parkinson, D., Nico, P., Ajo-Franklin, J., MacDowell, A., Kocar, B., Bethel, W., Sethian, J.: Statistical segmentation and porosity quantification of 3d x-ray microtomography. In: *SPIE Optical Engineering Applications*. pp. 813502–813502. International Society for Optics and Photonics (2011)
18. Waggoner, J., Zhou, Y., Simmons, J., De Graef, M., Wang, S.: 3d materials image segmentation by 2d propagation: a graph-cut approach considering homomorphism. *IEEE TIP* 22(12), 5282–5293 (2013)
19. Zhang, J., Tan, T.: Brief review of invariant texture analysis methods. *Pattern Recognition* 35(3), 735–747 (2002)